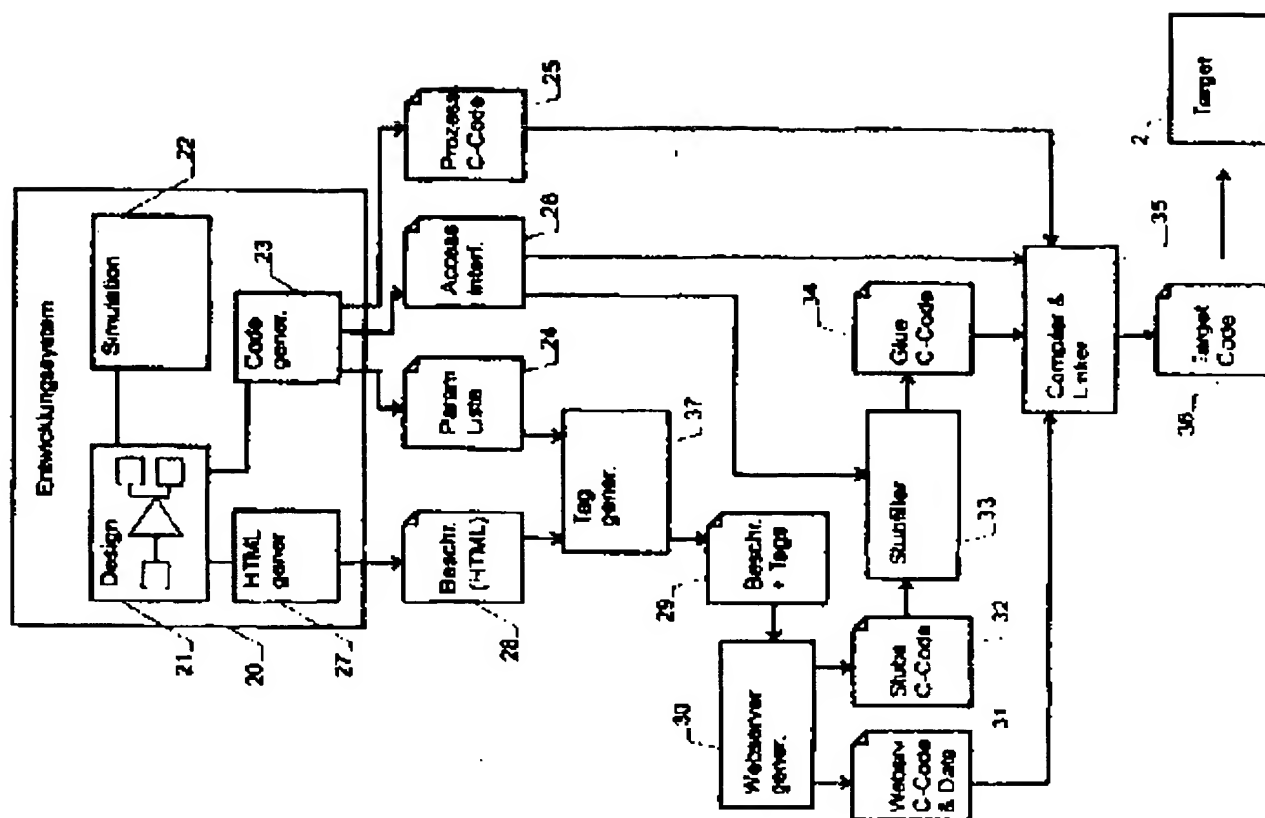


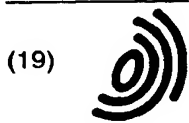
AN: PAT 2002-173132  
TI: Manufacturing regulator or controller involves generating web server to run on target system and access process code parameters to generate process control document in HTML/XML format  
PN: **EP1176482-A1**  
PD: 30.01.2002  
AB: NOVELTY - The method involves simulating the process on a development system and generating process code to run on the target system, generating a descriptive data file from the development system and a web server or data for one from the data file or from data derived from it. The web server can run on the target system and access parameters of the process code running on the target system to generate a process control document in HTML and/or XML format. DETAILED DESCRIPTION - The method involves simulating the process on a development system and generating process code in the development system for running on the target system. It involves generating a descriptive data file (28,29) from the development system and generating a web server or data for a web server from the data file or from data derived from it, whereby the web server can run on the target system and can access the parameters of the process code running on the target system to generate a process control document in HTML and/or XML format containing parameter input and/or output elements. INDEPENDENT CLAIMS are also included for the following: a computer program product with a program code arrangement.; USE - For manufacturing a regulator or controller with a regulation or control process running on the target controlled system and controlled by one or more parameters. ADVANTAGE - Enables a certain problem of conventional arrangements to be overcome in a simple manner, that is the need for expensive and extensive programs to be installed on the client computer. DESCRIPTION OF DRAWING(S) - The drawing shows a flow diagram representation of an example implementation of the method (Drawing includes non-English text) HTML and/or XML generator 27 descriptive data file 28,29 web server generator 30  
PA: (ALLM ) ABB RES LTD; (CREV/) CREVATIN M;  
IN: CREVATIN M;  
FA: **EP1176482-A1** 30.01.2002; US2002066072-A1 30.05.2002;  
CO: AL; AT; BE; CH; CY; DE; DK; EP; ES; FI; FR; GB; GR; IE; IT; LI; LT; LU; LV; MC; MK; NL; PT; RO; SE; SI; US;  
DR: AL; AT; BE; CH; CY; DE; DK; ES; FI; FR; GB; GR; IE; IT; LI; LT; LU; LV; MC; MK; NL; PT; RO; SE; SI;  
IC: G05B-019/042; G05B-019/418; G06F-009/44;  
MC: T01-J07B1; T01-J20B1; T01-N03B2; T01-S03; T06-A04B1; T06-A04B7;  
DC: T01; T06;  
FN: 2002173132.gif  
PR: EP0810676 27.07.2000;  
FP: 30.01.2002  
UP: 25.06.2002

**BEST AVAILABLE COPY**

**THIS PAGE BLANK (USPTO)**



**THIS PAGE BLANK (USPTO)**



Europäisches Patentamt  
European Patent Office  
Office européen des brevets



(11)

EP 1 176 482 A1

(12)

## EUROPÄISCHE PATENTANMELDUNG

(43) Veröffentlichungstag:  
30.01.2002 Patentblatt 2002/05

(51) Int Cl.7: G05B 19/042, G05B 19/418

(21) Anmeldenummer: 00810676.7

(22) Anmeldetag: 27.07.2000

(84) Benannte Vertragsstaaten:  
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU  
MC NL PT SE  
Benannte Erstreckungsstaaten:  
AL LT LV MK RO SI

(72) Erfinder: Crevatin, Mario  
8409 Winterhur (CH)

(74) Vertreter: ABB Patent Attorneys  
c/o ABB Business Services Ltd. (SLE-I)  
Haselstrasse 16/699  
5401 Baden (CH)

(71) Anmelder: ABB RESEARCH LTD.  
8050 Zürich (CH)

### (54) Verfahren und Computerprogramm zum Herstellen einer Regelung oder Steuerung

(57) Zur Herstellung einer Steuerung bzw. Regelung (1) wird zuerst ein Steuer- bzw. Regelprozess (3) auf einem Entwicklungssystem (20) entworfen und simuliert. Sodann wird der Prozess-Sourcecode (25) für ein in der Steuerung bzw. Regelung (1) angeordnetes Target-System (2) generiert, sowie eine Beschreibungsdatei (28, 29) im HTML-und/oder XML-Format, die unter anderem die Parameter für die Steuerung bzw. Regelung (1) beschreibt. In die Beschreibungsdatei (28, 29) werden Tags eingesetzt, die die Parameter kennzeichnen. Sodann wird die Beschreibungsdatei (28, 29) in einem Webserver-Generator (30) in Daten und Sourcecode (31, 34) für einen Webserver (5) umgewandelt, der auf dem Target-System (2) laufen kann. Dabei dienen die Tags dem Webserver-Generator (30) und/oder Webserver (5) zur Einfügung von Ein- und Ausgabeelementen, über die die jeweiligen Parameter gesetzt bzw. ausgelesen werden können. Mit dem vorliegenden System kann in einfacher Weise ein Interface zur Steuerung des auf dem Target (2) laufenden Prozesses (3) erzeugt werden, welches Interface ohne spezielle Software von jedem Client-Computer (7) benutzt werden kann.

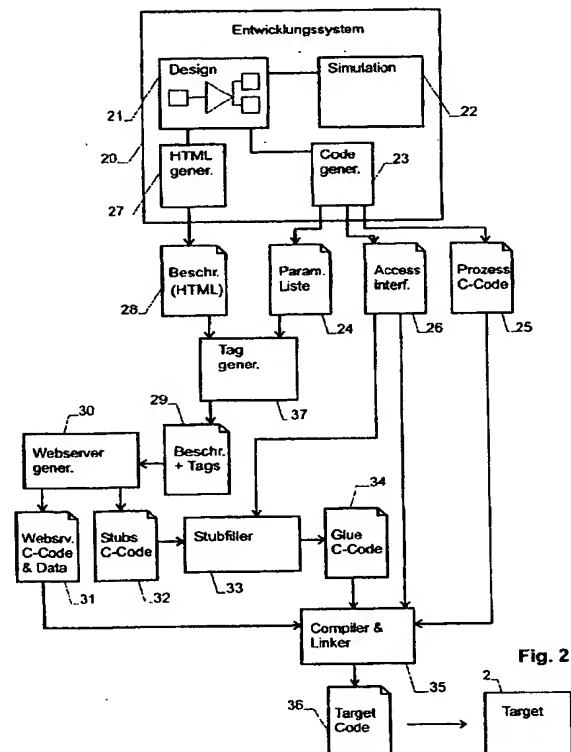


Fig. 2

EP 1 176 482 A1

## Beschreibung

### Technisches Gebiet

- 5 **[0001]** Die Erfindung betrifft ein Verfahren sowie ein Computerprogramm zum Herstellen einer Regelung bzw. Steuerung gemäss Oberbegriff der unabhängigen Ansprüche.

### Stand der Technik

- 10 **[0002]** Es gibt verschiedene Entwicklungssysteme, mit denen Steuerungen oder Regelungen entworfen und simuliert werden können, wie z.B. Matlab® und Simulink® der Firma The MathWorks, Inc., Natick (USA).  
**[0003]** Derartige Entwicklungssysteme erlauben es in der Regel, die entworfenen Steuerungen in C-Code umzuwandeln, so dass sie mit einem geeigneten Compiler kompiliert und als Prozess-Code auf einem Target-System in einer Regelung verwendet werden können.  
 15 **[0004]** Weiter bieten die Entwicklungssysteme die Möglichkeit, Parameter des auf dem Target-System laufenden Prozesses von einem Client-Computer aus zu prüfen und zu verändern. Hierzu müssen auf dem Client-Computer allerdings in der Regel teure und aufwendige Programme installiert werden.

### Darstellung der Erfindung

- 20 **[0005]** Es stellt sich deshalb die Aufgabe, ein Verfahren bzw. Software der eingangs genannten Art bereitzustellen, die es erlaubt, in einfacher Weise das obige Problem zu lösen.  
**[0006]** Diese Aufgabe wird vom Verfahren bzw. vom Computerprogramm gemäss den unabhängigen Ansprüchen gelöst.  
 25 **[0007]** Das Entwicklungssystem erzeugt also eine Beschreibungs-Datei, die zumindest die Parameter des Steuer- bzw. Regelprozesses beschreibt. Insbesondere enthält die Beschreibungs-Datei die Namen der Parameter, vorzugsweise auch den Typ bzw. die möglichen Werte der Parameter. Die Beschreibungs-Datei kann sodann von einem oder mehreren geeigneten Programmen dazu verwendet werden, einen Webserver oder Daten für einen Webserver zu generieren, der auf dem Target-System (d.h. der Steuerung bzw. Regelung) lauffähig ist und mindestens ein Prozesssteuerungs-Dokument in HTML-Format und/oder XML- beziehungsweise XML/XSL-Format bereitstellen kann, welches geeignete Ein- und/oder Ausgabeelemente für die Parameter enthält. Das Prozesssteuerungs-Dokument kann sodann von einem gängigen Browser von jedem Client-Computer abgerufen werden, so dass die Parameter des Prozesses in einfacher Weise ausgelesen bzw. geändert werden können.  
 30 **[0008]** Vorzugsweise gibt die Beschreibungs-Datei an, wie die Parameter im Prozesssteuerungs-Dokument darzustellen sind. Beispielsweise legt die Beschreibungsdatei die Reihenfolge oder die Orte fest, in bzw. an welchen die Parameter im Prozesssteuerungs-Dokument erscheinen sollen. Vorzugsweise ist die Beschreibungsdatei ebenfalls im HTML- oder im XML-Format, was die spätere Konversion vereinfacht.  
**[0009]** Vorzugsweise wird zusätzlich zur Beschreibungsdatei eine Parameterlisten-Datei erzeugt, die eine Liste der Parameter enthält. In der Regel sind gängige Entwicklungssysteme hierzu in der Lage. Sodann wird die Parameterlisten-Datei dazu verwendet, die Namen der Parameter zu ermitteln, und mit der so gewonnenen Information kann die Beschreibungsdatei durchforscht bzw. geparkt werden, um herauszufinden, wo die Parameter erscheinen. Die entsprechenden Orte werden sodann in geeigneter Weise mittels Steuerdaten, vorzugsweise mittels HTML-Tags, markiert. Dies ermöglicht es, aus den Dateien eines konventionellen Entwicklungssystem eine markierte Beschreibungsdatei zu erzeugen, die z.B. einem geeigneten Webserver-Generator zugeführt werden kann.  
 35 **[0010]** Vorzugsweise wird mit dem Entwicklungssystem eine Access-Interface-Datei erzeugt, die Deklarationen von Routinen enthält, über die auf die Parameter des laufenden Prozess-Codes zugegriffen werden kann. Diese Access-Interface-Datei wird bei der Erzeugung des Webservers zur Generierung von Code verwendet, mit welchem auf die Parameter zugegriffen werden kann. Dieses Vorgehen ist besonders für Entwicklungssysteme geeignet, die in der Lage sind, eine Access-Interface-Datei zu erzeugen.  
 40 **[0011]** Zum Erzeugen des Webservers wird vorzugsweise ein Webserver-Sourcecode erzeugt, der zusammen mit der Access-Interface-Datei kompiliert wird. Auch für den Prozess-Code wird vorzugsweise ein Prozess-Sourcecode erzeugt, der kompiliert und zum Webserver-Sourcecode gelinkt wird.  
**[0012]** Im Gegensatz zu bekannten Lösungen werden im erfindungsgemässen Verfahren mittels der Beschreibungs-Datei und gegebenenfalls der Parameterlisten-Datei Steuerdaten zur Verfügung gestellt, die zur automatisierten Erzeugung des Webservers benutzt werden, was einer der Beiträge der vorliegenden Erfindung auf technischem Gebiet ist. Auch erlaubt die vorliegende Art der Bereitstellung von Informationen das Zusammenwirken von Programmen (Entwicklungssystem, Webserver-Generator), die in der Regel mit inkompatiblen Datenformaten arbeiten.  
 45 **[0013]** Ebenso neu ist die Verwendung einer Access-Interface-Datei als Steuerdaten für einen Stubfiller, um eine

automatisierte Konversion oder Auffüllung der Stubs in Gluecode durchzuführen.

**[0014]** Das Verfahren wird vorzugsweise als Computerprogrammprodukt ausgestaltet, welches aus einem oder mehreren Programmen bzw. Modulen besteht und auf mindestens einem computerlesbaren Medium gespeichert oder auf einen Träger aufmoduliert ist.

#### Kurze Beschreibung der Zeichnungen

**[0015]** Weitere Ausgestaltungen, Vorteile und Anwendungen der Erfindung ergeben sich aus den abhängigen Ansprüchen und aus der nun folgenden Beschreibung anhand der Figuren. Dabei zeigen:

Fig. 1 eine schematische Darstellung der Komponenten bei der Entwicklung und beim Betrieb der Steuerung bzw. Regelung und

Fig. 2 ein Ablaufdiagramm mit den Programmkomponenten und Dateien bei einer möglichen Ausführung der Erfindung.

#### Wege zur Ausführung der Erfindung

**[0016]** Die wichtigsten Komponenten zur Entwicklung und zum Betrieb einer Regelung bzw. Steuerung 1 sind in Fig. 1 dargestellt.

**[0017]** Die Steuerung 1 besitzt ein Target-System 2, welches in der Lage ist, einen Regel- bzw. Steuerprozess 3, welcher als Prozess-Code vorliegt, auszuführen. Um seine Regel- bzw. Steuerungsaufgaben wahrzunehmen, kann der Regel- bzw. Steuerprozess 3 auf digitale oder analoge Ein- und Ausgänge 4 der Regelung bzw. Steuerung 1 zugreifen.

**[0018]** Auf dem Target-System 2 läuft weiter ein Webserver 5, der mit einer TCP/IP-Schnittstelle 6 verbunden ist. Der Webserver 5 ist in der Lage, ein oder mehrere HTML- und/oder XML-Dokumente (Prozesssteuerungs-Dokument 8) zu erzeugen, die die Parameter des Prozesses 3 darstellen bzw. Eingabeelemente für deren Änderung enthalten. Der Webserver 5 ist über die TCP/IP Schnittstelle 6 über eine Leitung oder ein beliebiges Netzwerk verbunden, über welche oder welches das bzw. die HTML- und/oder XML-Dokumente abgerufen werden können.

**[0019]** Das Target-System 2 kann als "embedded system" in einer einfachsten Ausführung ein Mikroprozessor-system ohne oder nur mit minimalem Betriebssystem sein, wobei in diesem Falle alle notwendige Software im Prozess 3 bzw. im Webserver 5 enthalten ist. Das Target-System 2 kann jedoch auch als Computer mit Betriebssystem ausgestaltet sein, welches dem Prozess 3 und dem Webserver 5 gewisse Dienste zur Verfügung stellt.

**[0020]** In Fig. 1 ist ein Client-Computer 7 dargestellt, mit welchem das Prozesssteuerungs-Dokument 8 über TCP/IP angefordert und in einem Web-Browser dargestellt werden kann. Im vorliegenden Fall enthält das Prozesssteuerungs-Dokument 8 z.B. eine graphische Darstellung der Steuerung bzw. Regelung in Form eines Blockdiagramms sowie eine Liste des bzw. der Parameter. In Fig. 1 ist z.B. ein Parameter mit dem Namen "gain\_x" vorgesehen. Er wird, zusammen mit seinem momentanen Wert ("2") im Prozesssteuerungs-Dokument 8 dargestellt. Der Wert erscheint in einem Eingabefeld 9, kann editiert werden und durch Anklicken eines Knopfs 10 an den Webserver 5 geschickt werden.

**[0021]** Wenn der Webserver 5 das Prozesssteuerungs-Dokument 8 erzeugt, muss er zuerst den Wert des bzw. der darzustellenden Parameter vom Prozess 3 abfragen. Wenn der Benutzer am Client-Computer 7 den Wert des Parameters ändert, so muss der Webserver 5 den neuen Wert an den Prozess 3 weitergeben. Hierzu ist eine geeignete Verknüpfung zwischen Webserver 5 und Prozess 3 notwendig.

**[0022]** Im folgenden wird beschrieben, wie eine Steuerung 1 und insbesondere die Software für den Webserver 5 und den Prozess 3 in effizienter Weise entwickelt und hergestellt werden können.

**[0023]** Zu diesem Zweck wird auf einem Entwicklungs-Computer 11 (oder gegebenenfalls mehreren in geeigneter Weise verknüpften Computern) zuerst die Steuerung bzw. Regelung entwickelt. Sodann wird der Prozess-Code für den Prozess 3 und der Webserver-Sourcecode oder Webserver-Code für den Webserver 5 erzeugt. Diese Codes sind auf dem Target-System 2 lauffähig und werden auf jede herzustellende Steuerung bzw. Regelung transferiert.

**[0024]** Eine bevorzugte Ausführung eines Ablaufdiagramms mit den Programmkomponenten und Dateien auf dem Entwicklungs-Computer 11 ist in Fig. 2 dargestellt.

**[0025]** In einem ersten Schritt der Entwicklung wird ein Entwicklungssystem 20 verwendet, welches erlaubt, die Regelung bzw. Steuerung zu entwerfen und zu simulieren. Hierzu sind ein Design-Modul 21 und ein Simulations-Modul 22 vorhanden.

**[0026]** Das Design-Modul 21 erlaubt es z.B., die Steuerung bzw. Regelung als Diagramm darzustellen und die Eingänge und Ausgänge sowie die Verknüpfungen der einzelnen Komponenten zu definieren. Ausserdem können die Parameter definiert werden, die das Verhalten der Steuerung bzw. Regelung festlegen. Als Parameter kommen z.B. Verstärkungsfaktoren, Schalterstellungen, Zeitkonstanten, Entscheidungsvariablen und andere Werte in Frage. Jeder

Parameter besitzt zumindest einen Namen und einen Typ (z.B. "double", "float", "Boolean"). Gegebenenfalls kann er noch weitere Attribute besitzen, wie z.B. eine Angabe darüber, ob er nur gelesen oder auch geändert werden kann.

**[0027]** Im vorliegenden Beispiel wird z.B. eine Steuerung bzw. Regelung definiert, die einen Eingang, einen Verstärker sowie zwei Ausgänge besitzt. Als Parameter ist der als "gain\_x" bezeichnete Verstärkungsfaktor des Verstärkers vorgesehen, der den Typ "double" besitzt.

**[0028]** Das Simulations-Modul erlaubt es, das Verhalten der so definierten Steuerung bzw. Regelung rechnerisch zu simulieren. Dies ermöglicht erste Tests der Steuerung bzw. Regelung.

**[0029]** Ist das Design soweit abgeschlossen, dass eine Steuerung bzw. Regelung 1 hergestellt werden kann, so muss der Code für den Prozess 3 erzeugt werden. Hierzu ist ein Code-Generator 23 vorgesehen. Dieser übernimmt die Design-Daten vom Design-Modul 21 und erzeugt drei Dateien bzw. Dateitypen: eine Parameterlisten-Datei 24, den Prozess-Sourcecode 25 und eine Access-Interface-Datei 26.

**[0030]** Die Parameterlisten-Datei 24 enthält eine Liste der Parameter. Insbesondere nennt sie für jeden Parameter den Namen und Typ. Im vorliegenden Beispiel enthält sie nur einen Eintrag, der z.B. so aussehen kann:

```
gain_x double
```

**[0031]** Der Prozess-Sourcecode 25 ist der Sourcecode für den Prozess 3. Er enthält die Routinen, um den Steuer- bzw. Regelprozess auf dem Target-System 2 auszuführen. Beispielsweise kann es sich um einen Sourcecode in C handeln. Der Prozess-Sourcecode enthält auch Accessor-Routinen, die es erlauben, die Parameter des Prozesses zu lesen bzw. zu verändern.

**[0032]** Die Access-Interface-Datei 26 enthält eine Deklaration der Accessor-Routinen, z.B. als Header File für die Programmiersprache C. Im obigen Beispiel könnte er z.B. folgende zwei Routinen definieren:

```
double read_gain_x(void); //lese gain_x
void write_gain_x(double); //setze gain_x
```

**[0033]** (In einer konkreten Ausführung sollten zusätzlich Fehlersituationen geprüft und entsprechende Fehlermeldungen ausgetauscht werden. So könnten die Funktion write\_gain\_x() z.B. eine Fehlermeldung zurückgeben, falls der gewünschte Parameterwert ausserhalb gewisser Grenzwerte liegt.)

**[0034]** Das Entwicklungssystem enthält weiter einen HTML- und/oder XML-Generator 27, der es erlaubt, die im Design-Modul 21 erzeugte Darstellung der Steuerung bzw. Regelung in ein HTML- und/oder XML-Format umzuwandeln und eine entsprechende Beschreibungsdatei 28 zu erzeugen, wobei hierunter auch ein Gruppe mehrerer verknüpfter Dokumente zu verstehen ist. Die Beschreibungsdatei 28 enthält z.B. als Link ein Diagramm der Steuerung bzw. Regelung sowie eine Liste der Parameter. Sie sieht, auszugsweise, z.B. so aus:

```
<HTML>
...
<IMG SRC="diagram.gif">
...
<P>gain_x: 1.0
...
</HTML>
```

**[0035]** Die soweit beschriebenen Komponenten des Entwicklungssystems 20 können z.B. konventionelle Programmprodukte sein. So kann insbesondere das eingangs genannte Matlab der Firma The MathWorks, Inc. verwendet werden. Dabei kann die Simulation in Matlab selbst durchgeführt werden. Als Design-Modul könnte Simulink® oder Stateflow® eingesetzt werden. Als Code-Generator 23 wäre "RealTime Workshop" bzw. "Stateflow Coder" zu verwenden, als HTML- und/oder XML-Generator "MATLAB Report Generator" und "Simulink", alle von der Firma The MathWorks, Inc.



**[0036]** Die Beschreibungsdatei 28 und die Parameterlisten-Datei 24 werden einem Tag-Generator 37 zugeführt. Der Tag-Generator 37 ist ein Computerprogramm oder Programm-Modul, und er führt in die Beschreibungsdatei 28 HTML-Tags ein, welche die Parameter kennzeichnen, so dass der Webserver 5 später die entsprechenden Ein- bzw. Ausgabeelemente in das Prozesssteuerungs-Dokument einfügen kann. Beispielsweise kann die Beschreibungsdatei folgendermassen umgewandelt werden.

```

<HTML>
...
<IMG SRC="diagram.gif">
...
<META
    NAME="parameter info"
    TEXT="gain_x:"
    PARAMETERNAME="gain_x"
    TYPE="double"
>
...
</HTML>

```

**[0037]** Der Meta-Tag ersetzt in der hier gezeigten Variante also die Zeile "<P>gain\_x: 1.0" und definiert, dass an der entsprechenden Position der Text "gain\_x:", das Eingabefeld 9 und der entsprechende Knopf 10 anzuordnen sind.

**[0038]** Nun wird die mit den Tags ergänzte Beschreibungsdatei 29 einem Webserver-Generator 30 zugeführt. Dieser erzeugt den Webserver-Sourcecode 31 und die Daten für den Webserver 5. Zudem erzeugt er eine Stub-Datei 32, die für jeden Parameter leere Routinen ("stubs") enthält. Diese Routinen werden später vom Webserver 5 aufgerufen, um die jeweiligen Parameter zu lesen und zu schreiben. Im obigen Beispiel kann die Stub-Datei 32 z.B. die folgenden Routinen definieren:

```

double GetParameter_gain_x(void)
{
}

```

```

void SetParameter_gain_x(double)
{
}

```

**[0039]** Zum Erzeugen des Webserver-Sourcecodes und der Stub-Datei 32 wird vorzugsweise ein konventionelles Produkt verwendet, beispielsweise EmWeb™ der Firma Agranat Systems, Inc., Maynard, USA.

**[0040]** In einem nächsten Schritt wird die Stub-Datei 32 und die Access-Interface-Datei 26 einem Stubfiller 33 zugeführt. Der Stubfiller 33 kann z.B. als eigenständiges Programm bzw. Programmmodul ausgestaltet sein. Er setzt in jede Routine der Stub-Datei 32 die entsprechende Routine der Access-Interface-Datei 26 ein und erzeugt eine Glue-Datei mit Glue-Sourcecode 34. Im vorliegenden Beispiel könnte diese z.B. vereinfacht so aussehen:

```
double GetParameter_gain_x(void)
```

```
{
    return read_gain_x();
}
```

```
void SetParameter_gain_x(double x)
```

```
{
    write_gain_x(x);
}
```

**[0041]** (Auch hier ist vorzugsweise noch Fehlerbehandlungs-Code hinzuzufügen.)

**[0042]** Nun können der Webserver-Sourcecode 31, der Glue-Sourcecode 34, die Access-Interface-Datei 26 und der Prozess-Sourcecode 25 in einem Compiler/Linker 35 kompiliert und gelinkt werden, um eine Codedatei 36 mit dem auf dem Target-System 2 lauffähigen Code für den Prozess 3 und den Webserver 5 zu erzeugen. Dieser Code kann auf das Target-System 2 transferiert werden.

**[0043]** Einmal auf dem Target-System 2 installiert, stellt der Webserver 5 auf Aufforderung dem Client-Computer 7 das Prozesssteuerungs-Dokument 8 zur Verfügung. Im vorliegenden Beispiel könnte es z.B. wie folgt aussehen:

```
<HTML>
```

```
...
```

```
<IMG SRC="diagram.gif">
```

```
...
```

```
<FORM METHOD="GET", ACTION="/bin/enter">
```

```
<P>gain_x:
```

```
<INPUT TYPE="TEXT" NAME="gain_x">
```

```
<INPUT TYPE="submit" VALUE ="OK">
```

```
</FORM>
```

```
...
```

```
</HTML>
```

**[0044]** Im obigen Beispiel sind das Entwicklungssystem 20, der Tag-Generator 37, der Webserver-Generator 30 und der Stubfiller 33 separate Programme bzw. Programmgruppen. Diese Abgrenzung ist jedoch nicht notwendig. So kann z.B. der HTML- und/oder XML-Generator 27 so ausgestaltet werden, dass er die für den Webserver-Generator 30 notwendigen Tags direkt in die Beschreibungsdatei einsetzt, so dass der Tag-Generator 37 entfallen kann. Alternativ hierzu könnte die Rolle des Tag-Generators 37 auch vom Webserver-Generator 30 übernommen werden. Auch der Stubfiller 33 könnte in den Webserver-Generator 30 oder den Code-Generator 23 integriert werden.

**[0045]** Im obigen Beispiel besitzt die Beschreibungs-datei vorzugsweise HTML-Format. Sie kann zusätzlich hierzu oder alternativ hierzu den XML-Formatierungsregeln folgen. In diesem Fall können anstelle der Tags auch andere, XML-Kompatible Datenstrukturen zur Steuerung des Webserver-Generators 30 in die Beschreibungsdatei eingefügt werden.

**[0046]** In den obigen Beispielen erscheinen die Parameter als Liste am Ende des Prozesssteuerungs-Dokuments 8. Es ist jedoch auch denkbar, ihre Werte direkt in das Diagramm einzusetzen, welches im oberen Teil des Dokuments erscheint. In diesem Falle ist der HTML- und/oder XML-Generator 27 so anzupassen, dass er den dazu nötigen Platz im Diagramm schafft, und es ist z.B. denkbar, dass der Webserver 5 ein Java-Applet erzeugt, dass Klickvorgänge auf

die im Diagramm erscheinenden Parameter mit entsprechenden Eingabemasken beantwortet.

[0047] Die Tags in der Beschreibungsdatei 29 werden im obigen Beispiel vom Webserver-Generator 30 interpretiert. Es ist jedoch auch denkbar, dass sie erst später vom Webserver 5 ausgewertet und in entsprechende Eingabe- und Ausgabeelemente umgesetzt werden.

[0048] In der Ausführung gemäss Fig. 2 werden die Code-Teile, wie z.B. der Prozess-Sourcecode, der Webserver-Sourcecode und der Glue-Sourcecode zuerst als SourceCode erzeugt und sodann kompiliert und gelinkt. Es ist jedoch denkbar, dass mindestens teilweise direkt lauffähiger Code erzeugt wird, der sodann nur noch gelinkt werden muss.

#### Bezugszeichenliste

[0049]

- 1: Steuerung
- 2: Target-System
- 3: Regel- bzw. Steuerprozess bzw. Prozesscode
- 4: Ein- und Ausgänge der Regelung bzw. Steuerung 1
- 5: Webserver
- 6: TCP/IP-Schnittstelle
- 7: Client-Computer
- 8: Prozesssteuerungs-Dokument
- 9: Eingabefeld
- 10: Knopf
- 11: Entwicklungs-Computer
- 20: Entwicklungssystem
- 21: Design-Modul
- 22: Simulations-Modul
- 23: Code-Generator
- 24: Parameterlisten-Datei
- 25: Prozess-Sourcecode
- 26: Access-Interface-Datei
- 27: HTML- und/oder XML-Generator
- 28: Beschreibungsdatei
- 29: Beschreibungsdatei mit Tags
- 30: Webserver-Generator
- 31: Webserver-Sourcecode
- 32: Stub-Datei
- 33: Stub-Filler
- 34: Glue-Sourcecode
- 35: Compiler/Linker
- 36: Codedatei
- 37: Tag-Generator

#### Patentansprüche

1. Verfahren zum Herstellen einer Regelung bzw. Steuerung (1), welche einen durch einen oder mehrere Parameter kontrollierten, auf einem Target-System (2) der Regelung (1) laufenden Regel- bzw. Steuerprozess (3) aufweist, wobei der Regel- bzw. Steuerprozess (3) auf einem Entwicklungssystem (20) simuliert wird und mit dem Entwicklungssystem (20) zum Ausführen auf dem Target-System (2) bestimmter Prozess-Code (3) erzeugt wird, und wobei das Verfahren **gekennzeichnet ist durch** die folgenden Schritte:

Erzeugen einer Beschreibungs-Datei (28, 29) aus dem Entwicklungssystem (20), wobei die Beschreibungs-Datei (28, 29) zumindest die Parameter beschreibt,

Erzeugen eines Webserver (5) oder von Daten für einen Webserver (5) aus der Beschreibungs-Datei (28, 29) oder aus Daten, die von der Beschreibungs-Datei (28, 29) abgeleitet wurden, wobei der Webserver (5) auf dem Target-System (2) lauffähig ist und ausgestaltet ist, auf die Parameter des auf dem Target-System (2) laufenden Prozess-Codes (3) zuzugreifen und ein Prozesssteuerungs-Dokument (8) in HTML- und/oder XML-Format zu erzeugen, welches Ein- und/oder Ausgabeelemente für die Parameter enthält.

2. Verfahren nach Anspruch 1, **dadurch gekennzeichnet, dass** die Beschreibungs-Datei (28, 29) Angaben zur Anordnung der Parameter im Prozesssteuerungs-Dokument (8) macht.

3. Verfahren nach Anspruch 2, **dadurch gekennzeichnet, dass** die Beschreibungsdatei (28, 29) in HTML- und/oder XML-Format ist.

4. Verfahren nach einem der Ansprüche 2 oder 3, **gekennzeichnet durch** die folgenden Schritte

Erzeugen einer Parameterlisten-Datei (24) mit dem Entwicklungssystem (20), wobei die Parameterlisten-Datei (24) für jeden Parameter zumindest einen in der Beschreibungs-Datei (28, 29) verwendeten Namen enthält, Erzeugen von Steuerdaten in die Beschreibungs-Datei (28, 29) unter Verwendung der Parameterlisten-Datei (24), wobei die Steuerdaten die in der Beschreibungs-Datei (28, 29) erscheinenden Parameter kennzeichnen, und insbesondere wobei die Steuerdaten HTML- und/oder XML-Tags sind.

5. Verfahren nach einem der vorangehenden Ansprüche, **dadurch gekennzeichnet,**

**dass** mit dem Entwicklungssystem (20) mindestens eine Access-Interface-Datei (26) mit Deklarationen für Routinen zum Zugriff auf die Parameter des laufenden Prozess-Codes (3) erzeugt wird und **dass** die Access-Interface-Datei (26) bei der Erzeugung des Webservers (5) für die Erzeugung von Code zum Zugriff auf die Parameter benutzt wird.

6. Verfahren nach Anspruch 5, **dadurch gekennzeichnet, dass** mit einem Webserver-Generator (30) Webserver-Sourcecode (31, 34) für mindestens einen Teil des Webservers (5) erzeugt wird, und insbesondere dass der Webserver-Sourcecode (31, 34) zusammen mit der Access-Interface-Datei (26) kompiliert wird.

7. Verfahren nach Anspruch 6, **dadurch gekennzeichnet, dass** mit dem Webserver-Generator (30) mindestens eine Stub-Datei (32) mit leeren Routinen für den Zugriff auf die Parameter erzeugt wird und dass in einem Stubfiller (33) die in der Access-Interface-Datei (26) definierten Routinen in die in der Stub-Datei implementierten Routinen eingefügt werden, um einen Glue-Sourcecode (34) zu erzeugen.

8. Verfahren nach Anspruch 7, **dadurch gekennzeichnet, dass** für mindestens einen Teil des Prozess-Codes (3) Prozess-Sourcecode (25) erzeugt wird, wobei die Access-Interface-Datei (26) im Prozess-Sourcecode (25) implementierte Routinen beschreibt, und dass der Prozess-Sourcecode (25), der Webserver-Sourcecode (31), der Glue-Sourcecode (34) und die Access-Interface-Datei (26) kompiliert und miteinander gelinkt werden.

9. Computerprogrammprodukt mit Programmcode-Mitteln, die auf mindestens einem computerlesbaren Medium gespeichert oder auf einen Träger aufmoduliert sind, um das Verfahren nach einem der vorangehenden Ansprüche auszuführen, wenn das Computerprogrammprodukt auf mindestens einem Computer ausgeführt wird.

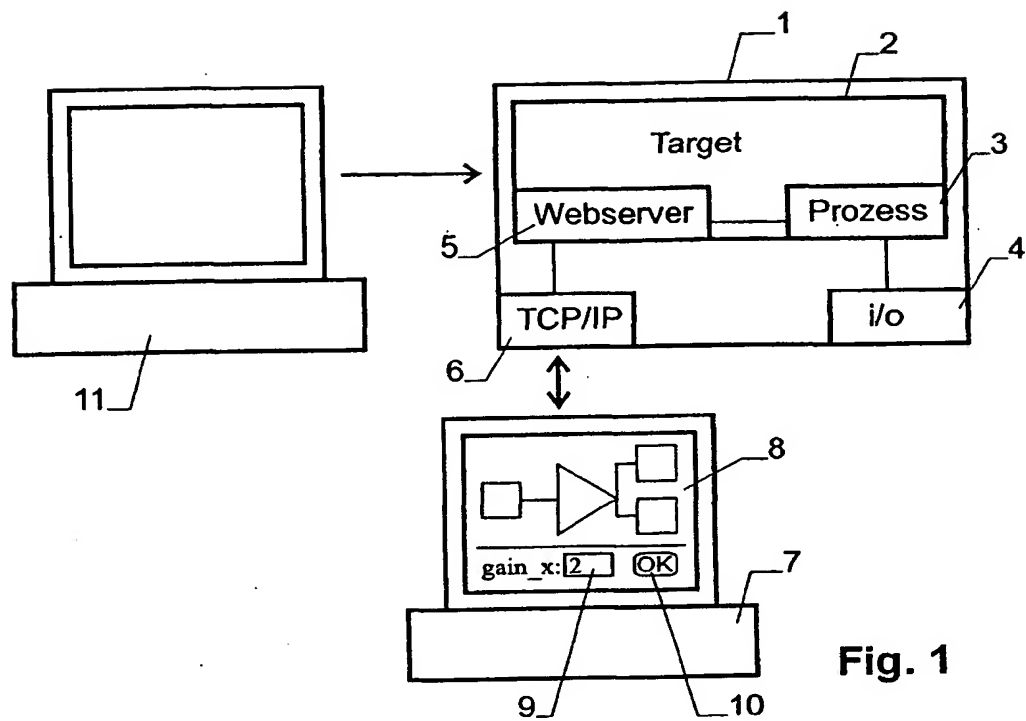


Fig. 1

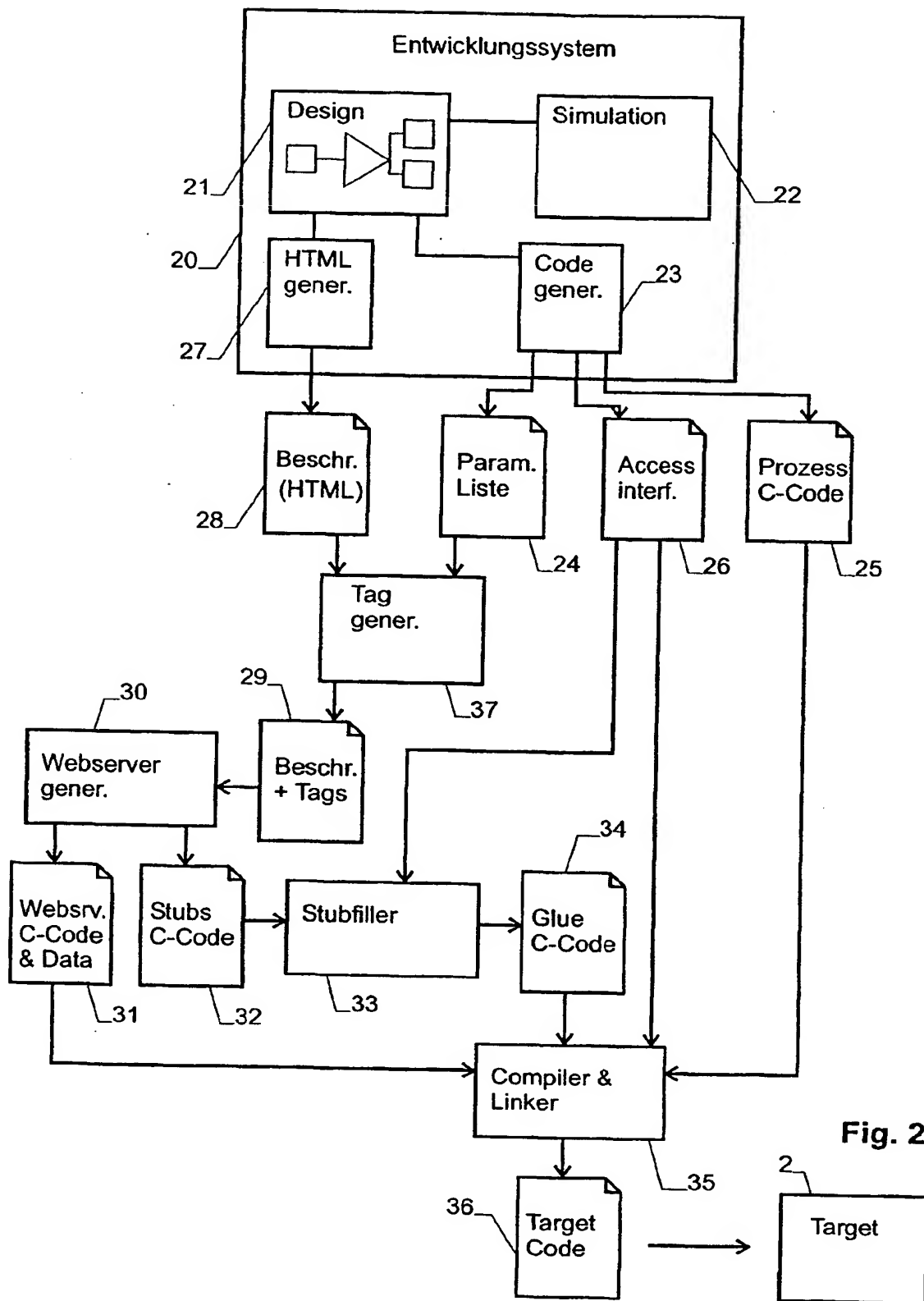


Fig. 2



Europäisches  
Patentamt

# EUROPÄISCHER RECHERCHENBERICHT

Nummer der Anmeldung  
EP 00 81 0676

EINSCHLÄGIGE DOKUMENTE			
Kategorie	Kennzeichnung des Dokuments mit Angabe, soweit erforderlich, der maßgeblichen Teile	Betrifft Anspruch	KLASSIFIKATION DER ANMELDUNG (Int.Cl.7)
A	WO 00 20939 A (PAVILION TECH INC) 13. April 2000 (2000-04-13) * Zusammenfassung * * Seite 7 - Seite 13 * * Abbildungen *	1-9	G05B19/042 G05B19/418
A	WO 97 12301 A (FISCHER HORST ; LOEWEN ULRICH (DE); SONST HORST (DE); FREITAG HARTM) 3. April 1997 (1997-04-03) * Zusammenfassung * * Seite 6, Zeile 17 - Seite 9, Zeile 17 * * Abbildungen 1-5 *	1,9	
A	WO 99 63409 A (CONTROL TECHNOLOGY CORP) 9. Dezember 1999 (1999-12-09) * Zusammenfassung * * Seite 12, Zeile 10 - Seite 18, Zeile 15 * Abbildungen *	1-5,9	
A	EP 0 825 506 A (FOXBORO CORP) 25. Februar 1998 (1998-02-25) * Zusammenfassung * * Spalte 3, Zeile 32 - Spalte 9, Zeile 1 * * Abbildungen *	1-3,5,9	RECHERCHIERTE SACHGEBIETE (Int.Cl.7) G05B G06F
A	REZAYAT M: "Knowledge-based product development using XML and KCs" COMPUTER AIDED DESIGN, GB, ELSEVIER PUBLISHERS BV., BARKING, Bd. 32, Nr. 5-6, Mai 2000 (2000-05), Seiten 299-309, XP004194679 ISSN: 0010-4485 * Zusammenfassung * * Seite 301 - Seite 302 * * Seite 304 - Seite 307 *	1-4,9	
-/--			
Der vorliegende Recherchenbericht wurde für alle Patentansprüche erstellt			
Recherchenort <b>DEN HAAG</b>		Abschlußdatum der Recherche <b>26. Januar 2001</b>	Prüfer <b>Hurtado-Albir, J</b>
KATEGORIE DER GENANNTEN DOKUMENTE X : von besonderer Bedeutung allein betrachtet Y : von besonderer Bedeutung in Verbindung mit einer anderen Veröffentlichung derselben Kategorie A : technologischer Hintergrund O : mündliche Offenbarung P : Zwischenliteratur		T : der Erfindung zugrunde liegende Theorien oder Grundsätze E : älteres Patentedokument, das jedoch erst am oder nach dem Anmeldedatum veröffentlicht worden ist D : in der Anmeldung angeführtes Dokument L : aus anderen Gründen angeführtes Dokument & : Mitglied der gleichen Patentfamilie, übereinstimmendes Dokument	

EPO FORM 1503 03 82 (P04C2)



Europäisches  
Patentamt

# EUROPÄISCHER RECHERCHENBERICHT

Nummer der Anmeldung  
EP 00 81 0676

EINSCHLÄGIGE DOKUMENTE			
Kategorie	Kennzeichnung des Dokuments mit Angabe, soweit erforderlich, der maßgeblichen Teile	Betrifft Anspruch	KLASSIFIKATION DER ANMELDUNG (Int.Cl.7)
A	<p>"XML-BASED TEMPLATES FOR GENERATING ARTIFACTS FROM JAVA-BASED MODELS"</p> <p>RESEARCH DISCLOSURE, INDUSTRIAL OPPORTUNITIES LTD. HAVANT, GB,</p> <p>Nr. 416, Dezember 1998 (1998-12), Seiten 1678-1680, XP000834144</p> <p>ISSN: 0374-4353</p> <p>* das ganze Dokument *</p> <p>-----</p>	1-4,9	
			RECHERCHIERTE SACHGEBIETE (Int.Cl.7)
Der vorliegende Recherchenbericht wurde für alle Patentansprüche erstellt			
Recherchenort <b>DEN HAAG</b>		Abschlußdatum der Recherche <b>26. Januar 2001</b>	Prüfer <b>Hurtado-Albir, J</b>
<b>KATEGORIE DER GENANNTEN DOKUMENTE</b> X : von besonderer Bedeutung allein betrachtet Y : von besonderer Bedeutung in Verbindung mit einer anderen Veröffentlichung derselben Kategorie A : technologischer Hintergrund O : mündliche Offenbarung P : Zwischenliteratur T : der Erfindung zugrunde liegende Theorien oder Grundsätze E : älteres Patentedokument, das jedoch erst am oder nach dem Anmeldedatum veröffentlicht worden ist D : in der Anmeldung angeführtes Dokument L : aus anderen Gründen angeführtes Dokument & : Mitglied der gleichen Patentfamilie, übereinstimmendes Dokument			

EPO FORM 1503 03/82 (P14/C03)



**ANHANG ZUM EUROPÄISCHEN RECHERCHENBERICHT  
 ÜBER DIE EUROPÄISCHE PATENTANMELDUNG NR.**

EP 00 81 0676

In diesem Anhang sind die Mitglieder der Patentfamilien der im obengenannten europäischen Recherchenbericht angeführten Patentedokumente angegeben.

Die Angaben über die Familienmitglieder entsprechen dem Stand der Datei des Europäischen Patentamts am 26-01-2001.  
 Diese Angaben dienen nur zur Unterrichtung und erfolgen ohne Gewähr.

26-01-2001

Im Recherchenbericht angeführtes Patentedokument		Datum der Veröffentlichung	Mitglied(er) der Patentfamilie		Datum der Veröffentlichung
WO 0020939	A	13-04-2000	AU	6294999 A	26-04-2000
			EP	1042706 A	11-10-2000
WO 9712301	A	03-04-1997	DE	19639424 A	27-03-1997
			EP	0852759 A	15-07-1998
WO 9963409	A	09-12-1999	AU	3975999 A	20-12-1999
EP 0825506	A	25-02-1998	KEINE		

EPC FORM P481

Für nähere Einzelheiten zu diesem Anhang : siehe Amtsblatt des Europäischen Patentamts, Nr. 12/82

**THIS PAGE BLANK (USPTO)**

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**

**THIS PAGE BLANK (USPTO)**